A comparison of human-computer user interface methods:
The effectiveness of touch interface compared to mouse

Andrew Muncey
MSc Information Systems
2014

Abstract

This dissertation examines the effectiveness of a touch user interface when compared with that of a traditional mouse. The effectiveness of a second hand, used to hold a touch interface is also considered.

Following an investigation into existing research in the domain of touch based user interfaces, an experiment was designed to evaluate the effectiveness of selection, dragging and gesture based input tasks undertaken with both a mouse and using a touch interface. Additionally operation of the touch interface when the device was held in the hand was compared to operation when the touch interface was situated horizontally on a desk, to determine the impact of bimanual operation.

The findings suggest that there is little variation in usability between a touch device held in the hand and situated on a desk, but that the touch interface provides an improved experience for an end user over that of a mouse based interface not only for selection as previous researches had indicated, but also for dragging and gesture interaction based input.

Disclaimer

This work is original and has not been previously submitted in support of any other course or qualification

Signed  Date

Dedication

To my daughter Chloe: the chances are if you ever come to read this, by the time you do there's a good chance the mouse may be obsolete.

Acknowledgments

I wish to thank my supervisor Andy Davies for all the help, guidance and encouragement he provided during the production of this work, and my colleagues within the Department of Computer Science, for the support they have given me both through their participation in this research and also throughout the first few years of my career in academia.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1. Background

Touch screen interfaces can go some way to alleviating issues caused by keyboard and mouse control by enabling the user to interact directly with the system's output rather than actions being translated through a peripheral. One of the earliest documented uses of touch screens is evidenced by Johnson (1967); Johnson and his colleagues had been developing a touch screen for evaluation and use within the Air Traffic Control Data-processing Systems in the UK. Johnson makes a statement that still applies to touch screens in the present day, that "only the computer programme affects the interpretation and labelling of the 'keys'" (p. 277). Touch screens were also developed at CERN in the early 1970s (Beck & Stumpe, 1973) in order to address the problem of "how to build the hardware for an 'intelligent' system which, in just three console units, would replace all those conventional buttons, switches etc." (CERN, 2010) Whilst CERN's first touch screen provided just nine buttons, modern touch screens can detect touch at arbitrary points on a screen. Touch screens allow users to interact with on screen elements controls such as buttons, hyperlinks and sliders, and in most ways still address that original problem – allowing control of an intelligent system where the display adapts to the end user's requirements.

Gesture recognition builds on the concept of touch recognition using software to recognise a series of consecutive touches. Software vendors such as Apple provide classes for gesture recognition of a number of predefined gestures such as swipe or pinch (Apple Inc., 2013c) however, given programmatic access to touch screen events, any conceivable touch gesture can be identified. Gesture controlled systems build on touch based systems by facilitating tasks that might otherwise require one or more button presses to be performed by recognising an end user's digit's movement across the interface.

This dissertation explores existing studies into the effectiveness of touch-based interfaces when compared to a mouse-based interface, looking at the input methods of tapping, dragging and a swipe gesture.

1.2. Research focus

There is limited research into the effectiveness of touch based interfaces, such as iPads and Android tablets when compared to the traditional point and click interface offered by a computer mouse. Researchers generally agree that for larger targets, and tap/click actions, a touch-based interface can be more efficient than a mouse, but that a mouse is quicker for drag events (Sears & Shneiderman, 1991) (Forlines, Wigdor, Shen, & Balakrishnan, 2007) (Cockburn, Ahlström, & Gutwin, 2012). Little work however has been published evaluating software interfaces designed primarily for touch, and pre-existing studies do not consider the supporting role of the secondary hand when carrying out touch-based tasks.

This dissertation firstly investigates previous research carried out in the field of touch and mouse based interface interaction and considers the role of the secondary hand in carrying out tasks, investigates the adaptation and design of user interfaces for touch interaction before undertaking empirical research to determine the utility of the second hand when using a touch based interface, and investigate the effective's of touch as a form of interface interaction when compared to a mouse.

Much of the focus of this dissertation is concerned with usability, and for the purpose of disambiguation, this is defined as "a quality attribute that assesses how easy user interfaces are to use" (Nielsen, 2012).

1.3. Aims and Objectives

This research and implementation had the following aims and objectives:

- To compare the effectiveness of completing identical tasks when using a touch screen compared to mouse input.
- To determine the perceived difficulty of tasks performed on both a touch interface and mouse interface.
- To determine if the bimanual use of a touchscreen (with one hand in a supporting role) elevates its usability above that of interaction using a single hand.

1.4. Scope and Limitations

The study was limited to evaluating touch, drag and swipe gesture events across mouse and touch screen interfaces. The number of participants was fairly small at eighteen, though this is comparable to the number of participants in other similar published studies.

1.5. Structure

This dissertation is structured in the following way:

Chapter 1 – Introduction
Chapter 2 – Literature Review
Chapter 3 – Methodology
Chapter 4 – Results and analysis
Chapter 5 – Discussion and Conclusion

2. Literature Review

This literature review discusses existing research and theory that compares mouse based interaction with that of touch, as well as gesture based systems and the adaptation of software for use on a touch based interface.

2.1. Previous comparative studies on mouse and touch interfaces

Sears and Shneiderman's (1991) study was one of the first to investigate the effectiveness of a touch screen interface. They first looked at the use of selection of varying sizes of target from 1-32 pixels per side (pixels having a non-square dimension of 0.4mm x 0.6mm). In order for the user to make the selection, a cursor was offset 6.1mm above the users finger, with an example of a similar technique shown in Figure 1. Software was written to stabilise the selection for one of the experiments. The targets reduced in size as participants completed the study, with the authors commenting that they wanted to "facilitate the subjects' skill acquisition" (Sears & Shneiderman, 1991, p. 7), and did so at the cost of using random ordering. The study showed that for selections of elements with sizes of 1.7x2.2mm and above, there was no difference in user performance. (The size of a target is known to affect the difficulty of acquiring it, and was originally identified by Fitts (1954).)

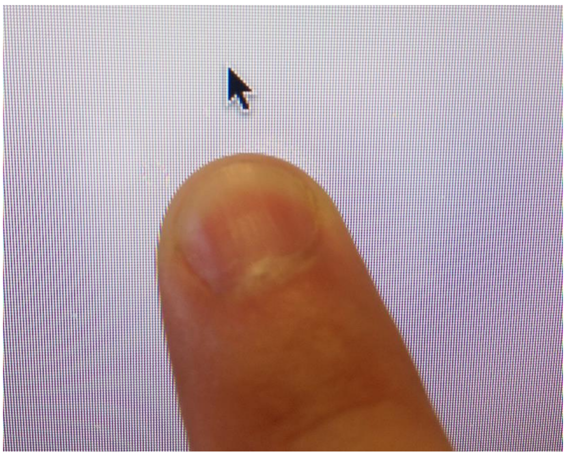

Figure 1 - Pointer offset above a digit

Forlines, Wigdor, Shen, & Balakrishnan's (2007) study compared the use of direct touch (i.e. no offset from digit to position of interaction on screen) on horizontal displays operated by twelve participants. Participants were required to select and drag items to be placed within 6mm (in this case 5 pixels) of the center of a target. The distances and target widths varied, but were repeated four times, with two blocks of experiments. Participants were asked about their preferred input device, with 75% choosing the mouse, but with minimal preference. They found touch interaction was slightly more efficient than mouse interaction but with a higher rate of error, and they identified that when larger targets were used the error rate decreased as would be expected considering Fitts law.

The second part of Forlines et al.'s study investigated the effectiveness of a bi-manual (two handed) approach, whereby participants attempted to use two mice simultaneously, or directly interacted with the screen using two hands simultaneously. There appears to be a slight flaw in their study if consideration is given to many other tasks where hands are used independently, albeit as part of the same task (such as playing the piano, typing or driving a vehicle) in that the skill required to complete such tasks proficiently is to be learnt through hours of practise, furthermore it typically requires the user to be able to conduct the task without hand-eye coordination.

Cockburn, Ahlström & Gutwin (2012) investigated performance in touch selections, investigating not only tap and drag interactions, but also radial pointing, and in addition to mouse and touch, they investigated the use of a stylus. The mouse-based system had a vertical display, whilst the touch based screen was horizontally positioned. In a similar manner to Sears et al., (1991) the pointer was offset from the finger position, but on both x and y-axis and by different values, as shown in Figure 2. Whilst the use of an offset between finger and area of interaction may enhance high precision selection tasks, it adds another layer of abstraction to an interface, and cannot be considered direct-touch.

They found that touch interactions were quicker than mouse for selection, but that mouse outperformed touch for drag events. Even though Cockburn et al.'s study took place in 2012 and so post dates the widespread adoption of touch interface tablet computing devices the participants in the study reported not having used a touchscreen more than a few times, however they were daily users of a mouse interface.

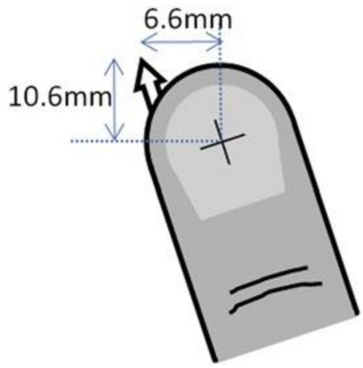

Figure 2 - Cursor offset (Cockburn et al., 2012, p.222)

2.1.1. Summary of studies

	Type of interaction	Participants	Method & Repetition	Cursor offset	Participant previous experience	Screen orientation
Sears and Shneidermann (1991)	Selection	36	12 tasks, 4 per input method	Vertical 6.1mm	Mouse: various Touchscreen: minimal/none	Vertical
Forlines, Wigdor, Shen and Balakrishnan (2007)	Dragging	12	1 task with variance. 2x4 repetitions for unimanual task, 8x16 for bimanual	None	Unknown	Horizontal
Cockburn, Ahlstrom and Gutwin (2011)	Selection, Dragging	18	1 task, 4 repetitions per input device	Drag only: vertical 10.6mm, horizontal 6.6mm	Mouse: regular Touchscreen: rarely/never	Horizontal

Table 1 - Summary of studies comparing touch and mouse interface interaction

2.1.2. Angle of operation

Forlines et al., p652 notes the issue of perspective distortion; in their study participants used a large horizontal screen, so as the user reaches to a more distant element of the screen, the angle between the digit and the interface becomes more acute, and therefore the element of a digit in contact with the screen changes from the tip of the digit (nearer the nail) to the flatter part of the end of the digit. They also noted that as the angle becomes more acute there is a potential for accidental input from other digits. (Incidentally the example given assumes that the index finger is the primary finger for pointing). The effect of perspective distortion can be minimised by angling a screen towards the user.

2.1.3. Bimanual interaction

Forlines, Wigdor Shen & Balakrishnan's study found that "for bimanual tasks… users benefit from direct-touch input", however in many tasks the secondary hand offers a supporting role for the primary hand, for example, to support a piece of wood being sawn by the primary hand, or to secure a saucepan whilst stirring the contents, whereas both hands were used in the study as primary actors in task completion. Guiard (1987, p. 494) found that when writing, the secondary hand supports and moves the paper for the primary hand, declaring, "Both hands contribute a truly dynamic component to the performance".

In the studies explored so far, the location of the touch screen has been fixed, thus removing the ability for the users secondary hand to contribute to the task in a supporting manner. This may be due in part to the weight and non-portability of touch-based interfaces in the past. Modern tablet devices are now significantly lighter than mouse driven devices, for example Apple's smallest and lightest laptop weighs 1.08kg with dimensions of 170mm x 192mm x 300mm (Apple Inc, 2013b) compared to the largest tablet which weighs 662g and has dimensions of 9mm x 185mm x 241mm. (Apple Inc, 2013a)

2.1.4. Prior use of touch interface

Participants in each study were either non-users, or non-regular users of a touch screen, and in most studies participants were regular users of a mouse interface. This leaves a potential for bias in the results in favour of the mouse-based interface.

2.1.5. Actions evaluated in the studies

The existing studies each used specially designed tasks, some elements of which were unlikely to correspond to actions that the participants may have been familiar with, for example the dragging of a rectangle to another position on screen. Whilst the operation of a button is likely to be familiar to the user, others actions may not be. Furthermore, some commonly used user interface elements such as drop down lists operate in a different manner, in particular making use of swipe gestures on touch devices. None of the experiments made use of such gestures.

2.1.6. Accuracy and efficiency

Forlines et al.'s study (2007, p. 649) required participants to "as quickly and as accurately as possible" carry out each task; they were prevented from continuing to ensure accuracy was not ignored in an attempt to progress quickly. Whilst Cockburn et al. (2012) do not describe how their participants were instructed to complete the task, the test system prohibited users from starting the tasks prior to the completion of a countdown timer.

2.2. Precise selections in modern touch screen implementations

Touchscreens by nature require the user to obscure at least part of the element of the user interface with which they are interacting. The technique used by Sears & Shneiderman (1991) offsetting the cursor position above the finger location has been adapted in modern touchscreen operating systems, for example, the iOS operating system provides a magnified view of the area under the user's finger above the selection (for accuracy in high precision selection techniques), if the user's digit persists in touching the same area of the screen, as shown in Figure 3. Whilst the digit cannot be seen from the screenshot, it is located above, and obscuring, the words 'precise' and 'selection', with the magnified image providing guidance for the user enabling them to carry out their task.

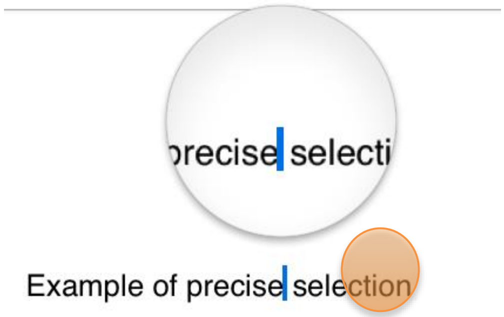

Figure 3 - Offset selection magnified in iOS (touch area in orange)

An alternative method of precise selection using a touch interface can be found in Microsoft's Remote Desktop application for iOS. (Microsoft Corporation, 2014). The software treats the touchscreen like a mouse pad, which moves a pointer on the screen in such a way that a movement of the finger translates to a much smaller movement of the pointer, making very precise interactions possible. In the example application, this is advantageous because the user is controlling a user interface designed for a desktop computer. Arguably however whilst the device being used is a 'touch-screen' the paradigm is broken through the increased abstraction.

2.3. Touch and gesture based systems

When comparing touch and gesture based systems against the desktop metaphor, it is often the task to be performed that dictates the preferable interface. Whilst achieving a level of accuracy of +/- 0.25mm is possible when interacting with a touch screen, including the capacitive type found in modern mobile devices (Atmel, 2013), due to the relatively large contact area that a human digit has with the screen, accuracy is generally much lower.

User interfaces cannot simply be replicated on a touch based interface; Subramanya and Yi (2006) note that "windows, icons, menus, and pointing (WIMP)—are inadequate and inappropriate for mobile applications". The original paper published by CERN in 1973 further identifies an advantage of the touch screen in providing only contextually relevant controls: "Not only can a few buttons perform in turn the functions of an arbitrarily large number, but there is no need to confuse the operator with the presence of buttons that are irrelevant in the current context." (Beck & Stumpe, 1973, p. 1).

The use of gesture recognition over and above touch recognition alone has multiple advantages. Gestures provide an additional form of input, without requiring any additional controls to be added to the screen visually, a common example would be swiping between pages in an electronic book. This can replicate an action that a user might perform in the real world, in this case dragging a finger over a page so that the page turns. A further advantage that Bragdon, Nelson, Li and Hinckley (2011) identify is that "gestures can be committed to muscle memory" and that this in turn helps the user "focus on their task".

In the above example of using a gesture to turn a page, the concept of replicating a real world object on screen is known as Skeuomorphism. Pogue (2013) explains that "Skeuomorphism in software has its place when used well: it can put you at ease with a new program in a flash". To some extent Skeuomorphism has been a common feature of computer user interfaces, with controls such as buttons using shading to present a three dimensional representation, imitating that of a physical button. Icons (for example files, folders and the recycle bin) extend the concept.

2.4. Adaptation of software for touch interfaces

Since the introduction of the iPad, a number of applications have been released for the platform that pre-existed for the desktop metaphor. This section evaluates some of those applications. The applications were selected as they offered very similar functionality across both mouse and touch interface versions.

2.4.1. GarageBand

Both OS X (desktop metaphor) and iPad versions of GarageBand enable the user to play and sequence music with various virtual instruments. (Apple Inc., 2013d) In the OS X version of the software, unless a physical digital musical instrument or controller is used (such as a midi capable keyboard), sounds can be input using the mouse to click on a virtual piano keyboard, or with a feature called musical typing. On the iPad however, a virtual piano keyboard can be played directly with fingers, furthermore the iPad adjusts the volume of the played note based on the velocity with which the piano key is pressed, a facility not available using keyboard or mouse click input. This touch sensitivity demonstrates an alternate method of recognising gestures, and a patent filed by Apple (Chin, 2010) explains how the accelerometer within a device can be used in conjunction with touches of the screen to identify the force of a touch gesture.

The iOS version of the GarageBand software also displays a number of visual differences in the user interface. Two notable differences between the application versions are the reduction in the number of controls displayed (buttons, sliders etc.) and the increase in size of the controls. A good example would be the volume control, in the OS X application; shown towards the bottom right of Figure 4 and top centre right of the iOS application shown in Figure 5.

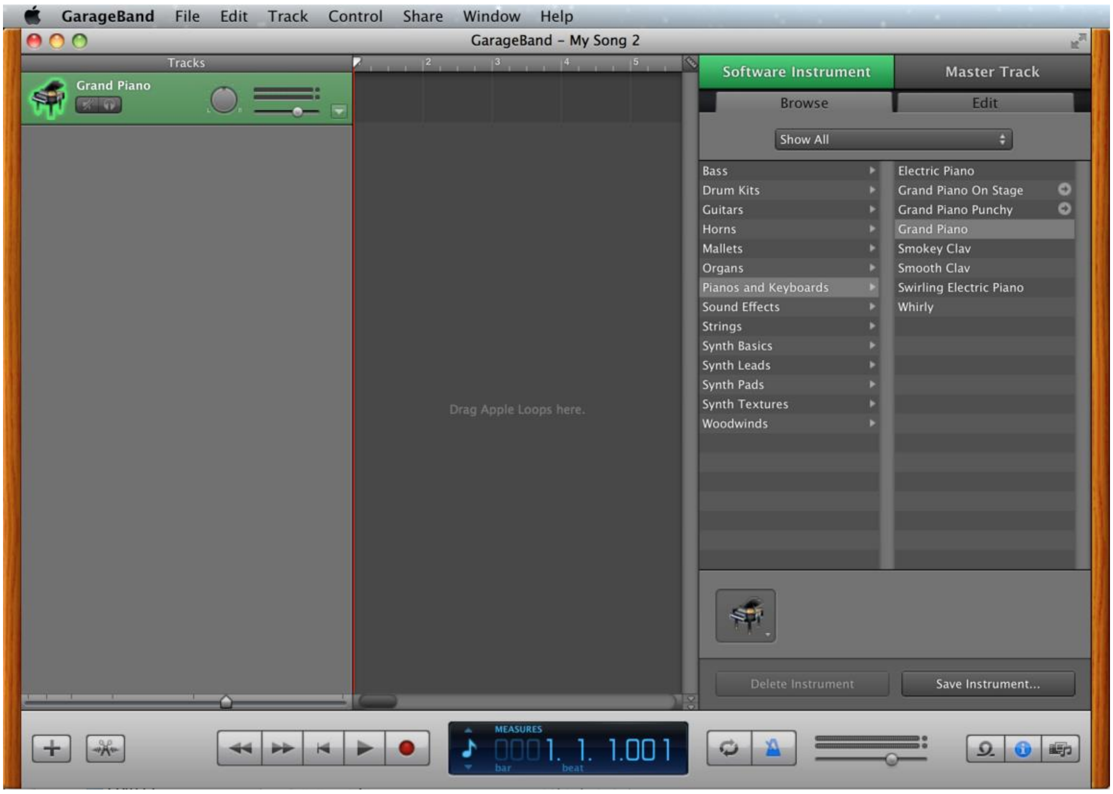

Figure 4 - GarageBand OS X Interface

Figure 5 - GarageBand iOS interface

The functionality to select a software instrument in the OS X user interface (Figure 4, right hand side) is not immediately visible within the iOS interface. This functionality has not been removed; rather it is accessible by pressing the instrument button in the centre of Figure 5. This in turn presents a modal view that splits the instruments into categories, seen in Figure 6. Whilst this requires one additional interaction with the application compared to the OS X version, larger controls allow the user to interact with them in a more efficient manner. (Fitts, 1954).

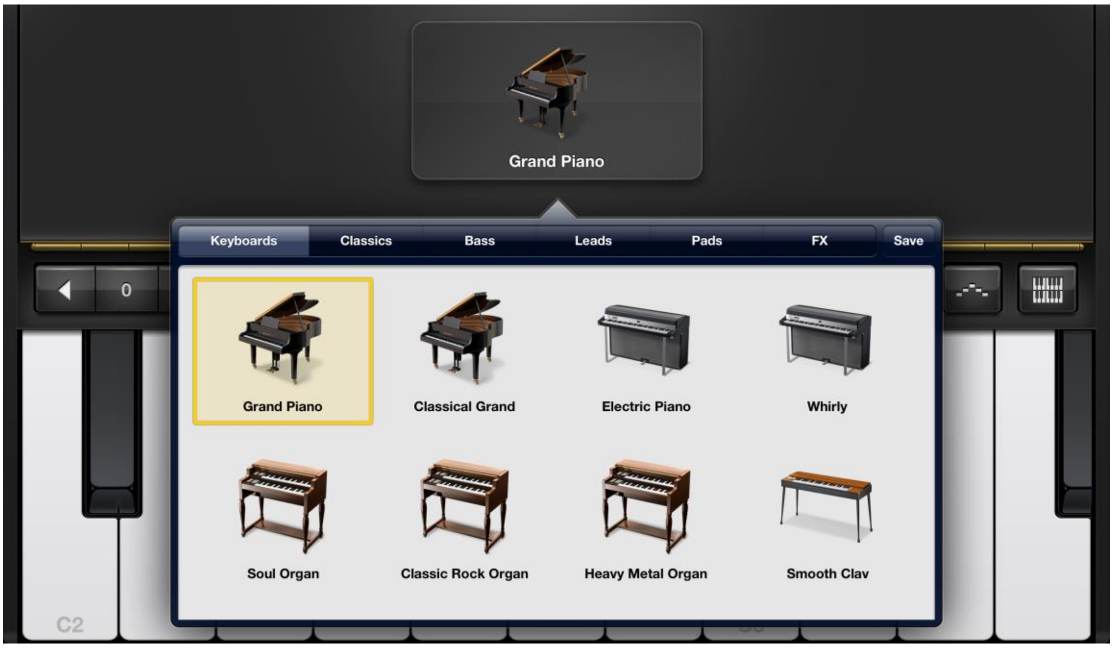

Figure 6 - Changing instruments in GarageBand for iOS

Other elements of functionality have been removed however, such as the fast-forward and rewind controls. These elements to some extent duplicate the functionality offered by the scrub tool, which again is enlarged in the iOS version of the app.

Additional changes include the ability to move the currently visible section of the keyboard using a pan gesture in the iOS version, compared to an additional window showing the active area of the keyboard in the OS X version, shown in Figure 7. Additional controls in the iOS version allow for switching between elements that are both visible in the OS X version, or to enable them to be presented in a modal view.

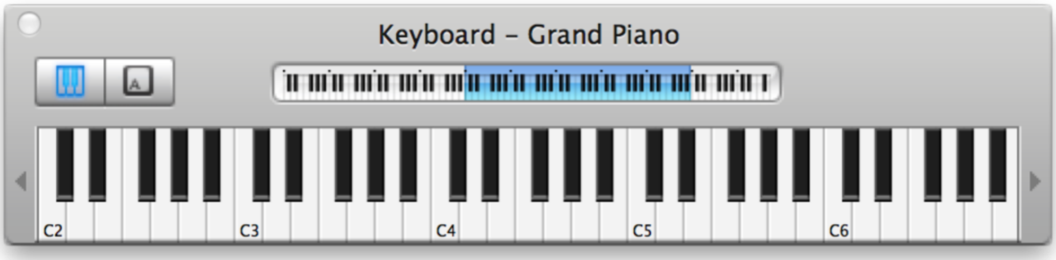

Figure 7 - Active keyboard selection tool in GarageBand for OS X

The GarageBand software demonstrates that a reduced number of controls, the removal of duplicate functionality, and controls that have a larger interaction area, are key considerations when adapting software for touch based devices.

2.4.2. Keynote

Keynote is software designed specifically for creating and displaying presentations, and Apple have released a version of the software for use on the iPad. This review concentrates on the presentation interface provided by the two software variants, as findings relate most closely to this project.

The desktop interface, shown in Figure 8 provides a minimal layout, showing the current slide, next slide, time of day and elapsed time. The four buttons shown at the top become visible only when the mouse moves towards the top of the screen. These buttons provide additional functionality, such as the ability to view and scroll through all slides and jump directly to one. Slides can be advanced and regressed with key presses and the mouse. Additional key presses, such as blanking the screen or hiding the presentation, provide further functionality. The display can be customised to show presenter notes, a countdown timer (instead of elapsed time) and a ready to advance indicator.

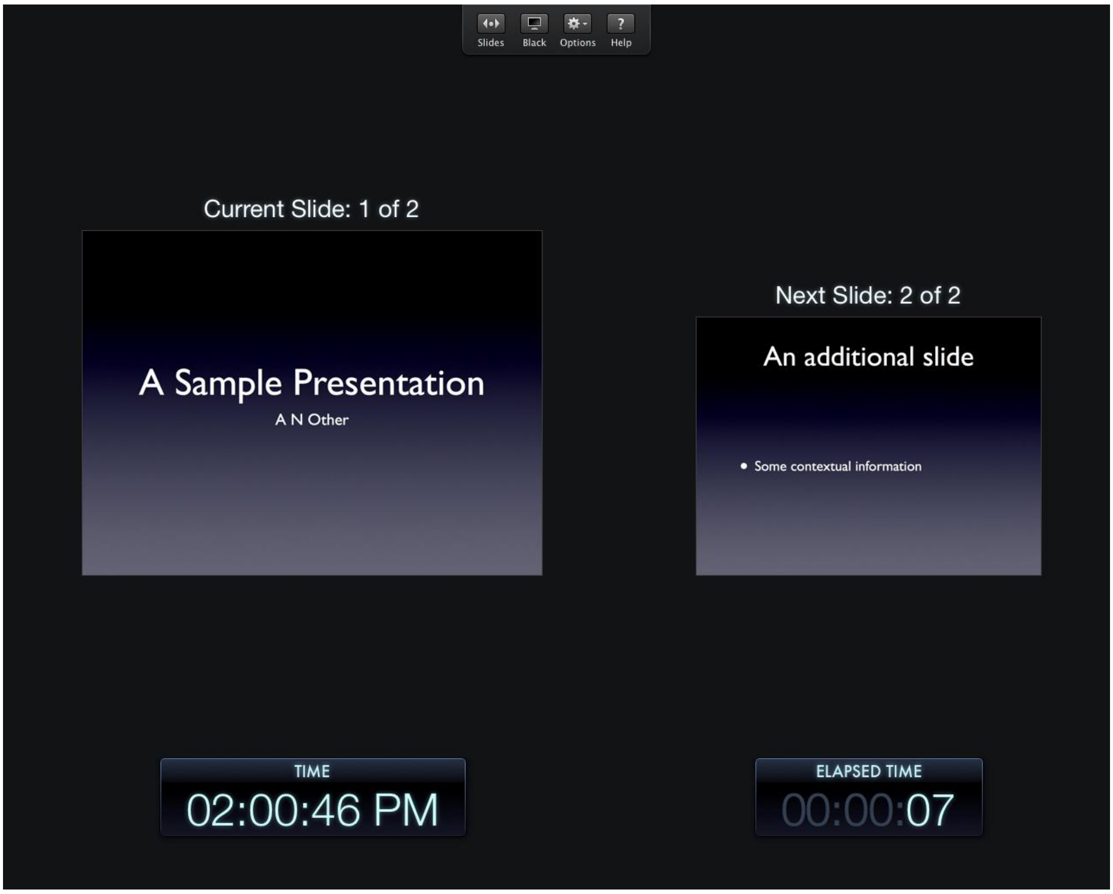

Figure 8 - Keynote presentation interface (OS X)

The iOS interface, shown in Figure 9 provides a minimal layout similar to the desktop interface, and permits the operator to switch between viewing the current, next, both current and next, or current and notes option. It should be noted that where the desktop interface permits notes as an additional on screen option when viewing multiple displays, on the touch interface, notes are an alternative to the next screen. The interface also provides the option to view all sides in the presentation in a modal scrolling view that appears to the left hand side of the screen, allowing the user to jump to any slide in the presentation. It should also be noted that slides often have multiple 'builds' or stages of animation, so this does not equate to being able to jump to any point in the presentation.

In addition to the changes in displayed features, controls in the iOS interface are noticeably larger than the OS X version.

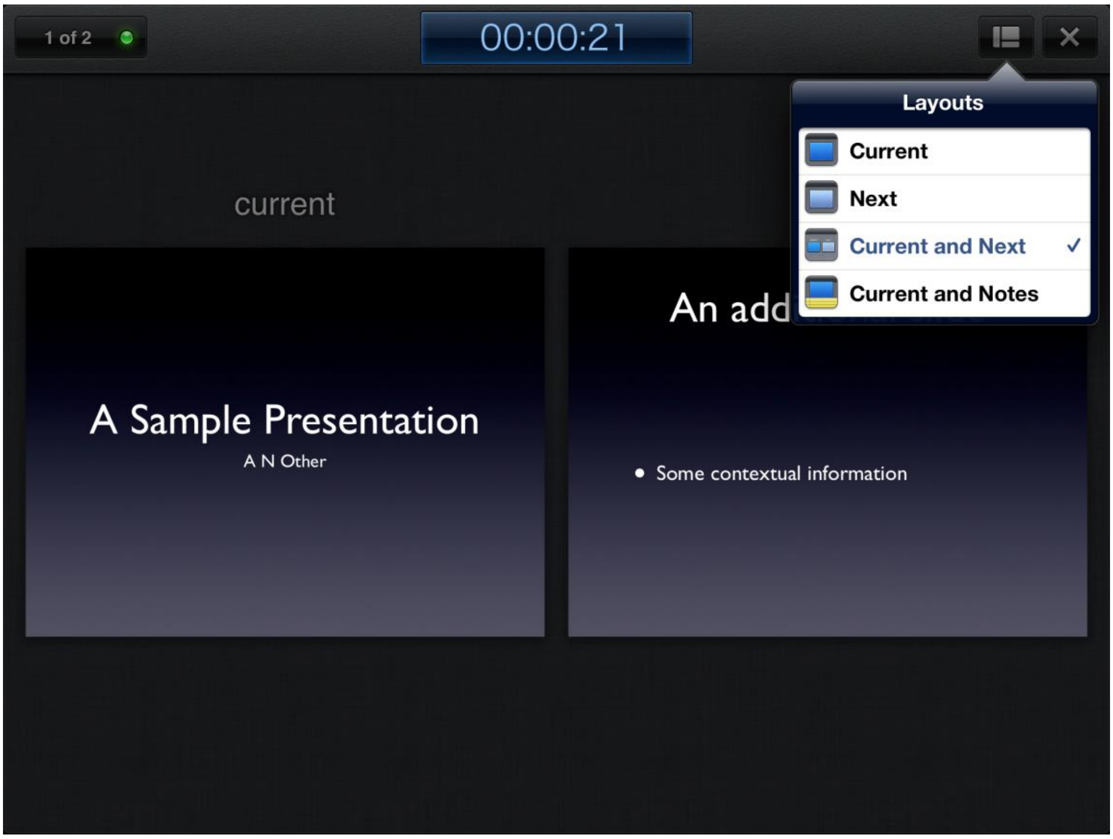

Figure 9 - Keynote Presentation interface (iOS)

The software adapted for touch interfaces in the examples reviewed above indicate a trend towards removing controls that duplicate functionality, and in some cases removing elements of functionality; controls are larger, and the number of controls displayed at any one time is reduced. Functionality that is directly visible in desktop software is often accessible though modal menus in the touch based software.

2.5. Conclusion

The existing studies into touch and mouse interfaces indicate a need for further research into interaction with touch-interfaces, in particular the consideration of bimanual interaction with the additional hand in a supporting role. Where possible the angle of interaction between the finger and screen should be consistent, users should be familiar with both types of interface, and research should check both accuracy and efficiency of the interaction.

Interfaces should be primarily designed for touch and gesture based interaction, as previous studies either designed bespoke content or used existing PC interfaces, which may cause inaccuracy due to the larger surface area of the finger as a pointing device. Controls should be large enough to be controlled by human digits. The literature indicates that the traditional concepts of windows, menus and pointers do not extend well to the touch interface and should be avoided, however well designed skeuomorphic controls can reduce the cognitive burden on the user. Furthermore, cognitive burden can be reduced further through the use of gestures, as they can be committed to muscle memory.

3. Methodology

3.1. Introduction

3.1.1. Aims

The aim of the experiment is to determine whether having the user hold a touch screen device has any impact on its usability compared to having it positioned on a desk and secondly that the usability of tasks that involve on screen selection, dragging and gesture controlled tasks, varies depending on whether the interface is touch or mouse based.

3.1.2. Null Hypotheses and Alternative Hypotheses

3.1.2.1. Hypothesis A – comparison of touch with a supporting second hand

H_0 The use of the second hand supporting a device does not affect the efficiency of the user's interaction with its touch-based interface.

H_1 Use of a touch-based interface with the secondary hand supporting the device affects the efficiency of interaction using a device situated on a desk.

3.1.2.2. Hypothesis B – comparison of mouse and touch

H_0 A touch-based interface for a system is no more or less usable than an interface for the same system which requires use of a mouse.

H_1 A touch-based interface for a system varies in its usability compared to an interface for the same system which requires use of a mouse.

3.1.3. Ethics

The University of Chester faculty research ethics committee for the Faculty of Life Sciences approved the empirical research undertaken and the application and approval can be found in appendix 1. The primary ethical issues were the slight potential for discomfort when operating either the mouse, or holding and operating the tablet device, and the cognitive burden on the participants as they were asked to complete tasks quickly. Participants were able to leave at any point and take breaks between tasks. Participants were linked to their results through a unique number and all data was anonymised prior to publication. All data that could identify a participant gathered was stored securely, with physical data in a locked storage unit and electronic data on an encrypted password protected computer.

Before starting the tasks participants were given an information sheet explaining the research and ethical issues, which can be found in appendix 2. Once they had read the participant information sheet they were asked to consent, and a sample consent form can be found in appendix 3.

The researcher adhered to the BCS code of good practice, in particular the guidance for performing research found in section 4.2. (British Computer Society, 2004, p. 20)

3.2. Research Strategy

3.2.1. Experiment design

The overall research strategy was experimental, as it allows the study to be repeated, and offers precise measurements, the elimination of alternative hypothesis and tight control over variables (Biggam, 2011). A parametric approach was taken, as there were a high number of instances of data being generated, with ratio variables being gathered in the form of timings from the software and interval values being taken from the completed questionnaires. The experiment used a within-group design with all participants being exposed to all experiment conditions. The order in which participants undertook the task was varied, with one third of participants starting with the mouse based task, one third holding the iPad and one third with the iPad on the desk. The order in which the three tasks were undertaken was also evenly distributed amongst participants, with the aim of negating any learning effect. Furthermore, in order to further reduce the learning effect participants were asked to practise the tasks until they were familiar and comfortable with them before starting the timed assessment. In order to reduce the effect of fatigue participants were permitted breaks between tasks and between elements of each task.

3.2.2. Dependent Variables

Two dependent variables were explored as part of the study: The user input method (either mouse or finger) and the placement of the touch device (either on a desk or held in one hand)

3.2.3. Sample

Love (2005, p. 41) argues that sample size may need to be as high as 30, though he cites usability expert Jakob Nielsen who argues that 5-10 participants is sufficient. The sample size used was 18. Due to budgetary and time constraints convenience sampling was used, and as a result the sampling was non-random, comprised of colleagues and other contacts of the researcher; it was further limited to participants who volunteered their time, as there was no incentive to participate. These factors were not expected to affect the results of this exploratory research.

3.2.4. Bias

Due to the relatively recent introduction of touch screen tablet computing devices, participants had significantly more experience using a mouse. It is hypothesised that this may have introduced a bias towards users interacting more efficiently with the mouse rather than the touchscreen. Until such time as potential participants have 'grown up' with touch screen technology being commonplace, this phenomenon will continue to be an issue within the adult population for a number of years. In order to reduce the incidence of random error, participants were permitted to rehearse the task in all cases, and for each interface method they repeated all aspects of the task eight times.

Observer bias was limited by ensuring calculations were made automatically using the software rather than by the observer, and a form of objective measurement was used for collecting participant responses.

Participant bias was limited in two ways, by ensuring that participants did so voluntarily, and also, as suggested by Mitchell and Jolley (2013, p. 158), by "making it costly", in this instance by them having to repeat the task as deliberately slowing down the speed they carry out the task wastes the participant's time. The nature of the task limits any bias on the grounds of social desirability.

3.3. Data Collection

Data was collected using two methods. Bespoke software was developed for the task, which accurately recorded timing data; the timing data was later exported in comma separated value format for analysis. This was designed to provide the ability to analyse and compare the empirical data once the data collection stage was completed. Participants completed the same task eight times under each of the three conditions (mouse, touch on static device, touch on held device, totalling 24 instances). After each of the individual experiment conditions, participants completed the NASA task load index questionnaire to gauge their perception of the task undertaken. Informal notes were also made while observing the participants.

It should be noted that when using the iPad participants were not instructed about whether or not they could use both hands for touch input (as opposed to supporting the device which was controlled). Use of bimanual input was observed and noted.

3.3.1. Choice of hardware

Whilst interaction with a touch screen is likely to vary only minimally by device, as a result of the coefficient of friction of the screen's (typically glass) surface, the design and ergonomic form of computer mice can vary significantly. In order to minimise the effect of potentially using an unfamiliar mouse, a popular wireless mouse was chosen. Popularity was determined using the popularity sort on the 'Mice' product page on the amazon.co.uk website. Whilst this is a rather crude method of determining popularity, comprehensive research into commonly used mice was outside the scope of this research. The mouse used was the Logitech M175 wireless mouse, and the decision to use a wireless mouse over that of a wired mouse was made to avoid the potential restriction of movement introduced by a mouse cord. The mouse was used directly on a laminate wood desk surface, thus avoiding the potential for having to 'clutch' the mouse, which can occur when a mouse mat is used and the user needs to move the cursor in such a way that the mouse would overrun the mouse mat.

It is important to note that cursor position is not directly proportional to mouse movement in modern computer operating systems, for example the Windows 7 operating system has an option to 'Enhance pointer precision' turned on by default. Microsoft explains that this option causes the "pointer [to] work more accurately when you're moving the mouse slowly" (Microsoft, 2014). Whilst there is no such setting explicitly visible within Mac OS X Mavericks' mouse settings, similar behaviour was observed by the author when using the mouse. Researchers claim that most studies indicate that such adaptive gain or attenuation of cursor position provides enhanced interface interaction (Blanch, Guiard, & Beaudouin-Lafon, 2004). In order for the study to accurately compare interfaces as they are typically used, the mouse task was completed with adaptive gain left in its default (enabled) state. The mouse tracking speed was also left at the default setting.

An iPad was used as the touch screen device due to the development environment offering a simulator which could be used with a mouse, and also due to its wide adoption among users and minimal variation between devices, raising the likelihood of participants being familiar with such a device. The computer used for testing was an Apple MacBook Pro running the OS X Mavericks operating system, with the use of the iOS simulator necessitating the use of the Apple computer.

3.3.2. Development of software

Software was developed to run on the Apple iPad, with the operating system of the iPad being iOS 7. The Xcode integrated development environment was used for writing software and the programming code was written in Objective-C. As the SDK provide the facility for an iPad application to be simulated on a computer and operated with a mouse, the same software was used to test the mouse interface. The full source code for the software can be found in appendix 6, excerpts from the code showing a few key functions are shown in Figures 10, 11 and 12 below.

```
- (IBAction)startButtonPressed:(UIButton *)sender {

    //log time
    _startTime = [NSDate date];
    self.button1.enabled = YES;
    [sender removeFromSuperview];
}
```

Figure 10 - Code to determine start time and commence selection task

```
-(void)end{
    NSTimeInterval interval = [[NSDate date] timeIntervalSinceDate:_startTime];

    TestMeasurements *results = [TestMeasurements sharedMeasurements];
    results.currentMeasurement.buttonPressTime = interval;

    [self performSegueWithIdentifier:@"toDrag" sender:self];
}
```

Figure 11 - Code to record and store timing data from selection task

```
-(NSString *)measurementAveragesAsCSVData{

    NSMutableString *csvString = [[NSMutableString alloc]init];

    double tap = 0;
    double gesture = 0;
    double drag = 0;

    for (TestMeasurement *m in self.completedMeasurements)
    {
        tap += m.buttonPressTime;
        gesture += m.scrollTime;
        drag += m.topLeftToBottomRightDragTime + m.topRightToBottomLeftDragTime +
        m.bottomLeftToTopRightDragTime + m.bottomRightToTopLeftDragTime;
    }

    tap = tap / self.completedMeasurements.count;
    gesture = gesture / self.completedMeasurements.count;
    drag = drag / self.completedMeasurements.count;
    drag = drag / 4.0;

    [csvString appendString:@"\rSelection,Drag,Gesture\r"];
    [csvString appendFormat:@"%f,%f,%f\r\r",tap,drag,gesture];

    [csvString appendString:[self.taskLoadIndex indexAsCSVData]];

    return  csvString;
}
```

Figure 12 - Partial code to average data and convert to CSV format

When designing the tasks, Nielsen's heuristics were considered (Nielsen, 1995), in particular the use of minimalist design – only the required elements for the task were made available to the user, and care was taken to ensure that the language used was natural and logical. Consistency in UI design was also implemented to ensure users could recognise, rather than have to recall how to interact with them. On screen instructions enabled the system to be used without documentation. One element of Nielsen's (1995) heuristics that was purposely not implemented was the use of accelerators, because of the requirement for the functionality to be tested using only one method of interaction.

Three distinct tasks were incorporated into the software, a selection (tap or click) task, dragging task and gesture task. Prior to starting the tasks the users were required to enter their participant number. The task recorded the interface option used automatically for the mouse based interaction, but users had to select manually if they were holding the iPad or using it on a desk. Users could also choose to complete the tasks in a practise mode, which inhibited data logging.

The tap task required users to select 5 buttons in sequence. Users initiated the tasks themselves by clicking on a 'Start' button. A screenshot of the application can be seen in Figure 13. Targets were 91 pixels[1] wide and 53 high (17.29mm x 10.07mm).

The software recorded the time at the point the user clicked the start button, and calculated the elapsed time once the final button was pressed. It should be noted that the recognition of touch in this case was when the finger or mouse lifts from the button (a common implementation designed to avoid accidental touches or mouse clicks).

The drag task required users to drag a yellow rectangle to within the bounds of a larger grey rectangle, a screenshot of the initial screen can be seen in Figure 14. The timing commenced as soon as the user first interacted with the yellow rectangle (i.e. the touch down onto the rectangle) and ceased once it had been placed within the bounds of the target rectangle. When displayed on the screen to be used with the mouse the margin of error permitted was 20 pixels in any direction from the centre position (i.e. the target rectangle was 40 pixels wider and higher than the moveable rectangle), this corresponded to an allowable margin of error of approximately 7.6mm on the iPad. The yellow rectangle was 204 pixels wide by 119 high (38.8mm x 22.6mm). The centres of the rectangles were offset from each other by 550 pixels horizontally and 700 pixels vertically in each case (resulting in a diagonal movement required of 84.8mm on the iPad). It should be noted that when the user started the drag task, the movable rectangle maintained its position relative to the cursor or finger, i.e. it did not re-centre itself under the cursor or finger.

[1] Pixels, when referred to in the context of this experiment consider the iPad as having a resolution of 768 pixels by 1024 pixels (as found on older iPad devices, and as used in the development environment). When the software was deployed an iPad with resolution of 1536 x 2048 pixels was used, effectively displaying twice the number of pixels on both axis, but with no effect on the experiment measurements.

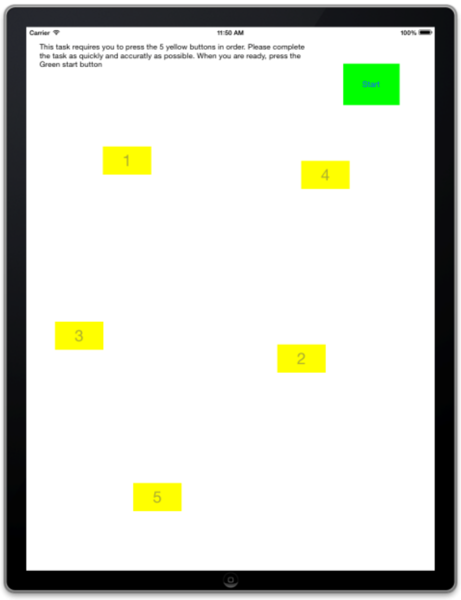

Figure 13 - Experiment tap/click task

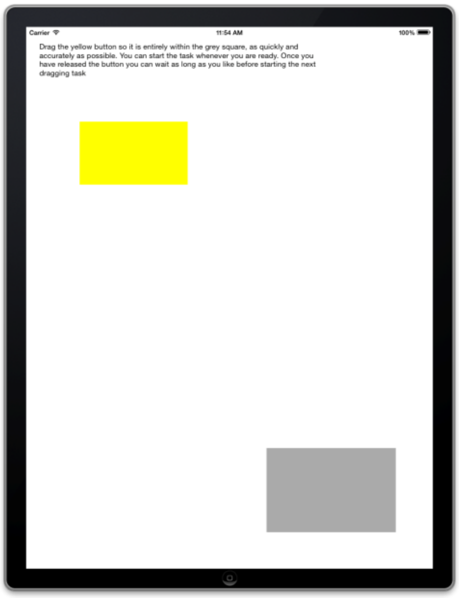

Figure 14 - Experiment drag task

The final task required the user to scroll through a list to select an item, this allowed the user to swipe or flick the list (in a similar manner to which one might spin a freely moving wheel by hand). The timing for this task started as soon as the list started to be moved, and ceased once they had correctly selected the desired item. The screen for this can be seen in Figure 15.

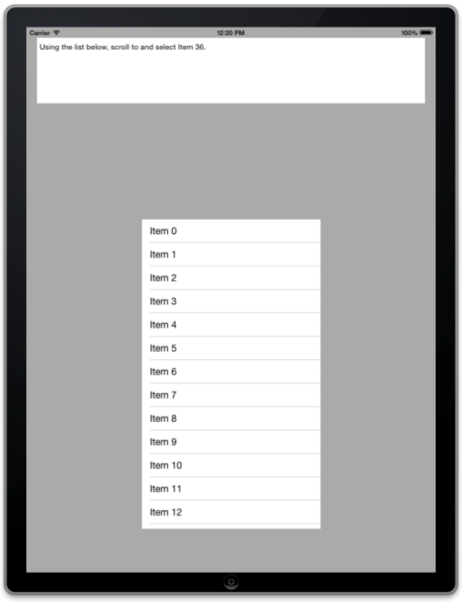

Figure 15 - Experiment gesture task

The application then stored the timing data on the device's memory (for later retrieval using iTunes file sharing), or in the case of the application running in the simulator, the file was saved for later retrieval from the computer's hard drive.

3.3.3. Pilot

A pilot study was carried out to identify any issues that would otherwise have occurred during the later stages of usability analysis. As a result of this, the software was modified so that data was sent only after all the repetitions of each task type (e.g. mouse, or screen) had been completed. Furthermore, a practise mode was added, so that data was not collected during test sessions.

3.3.4. Timing Data

The software collected timing and participant data for each task instance. This data comprised participant number, method of interaction (mouse, touch-screen on desk, touch screen in hand) the time taken to complete the selection task, the time taken for each of four dragging tasks (each corner of the screen to the diagonally opposing corner), and the gesture task.

3.3.5. Questionnaire

A number of techniques can be used in order to measure cognitive load on the user, with the use of a rating scale being most heavily adopted. Both Love (2005) and Neilsen (1993) argue that Likert-type usability questionnaires are suitable for measuring subjective satisfaction of a human computer interface, although alternative techniques such as measuring heart rate variability and pupillary response have also been used by researchers (Paas, Tuovinen, & Van Gervan, 2003). Of the scale based systems, in their 1992 paper "Comparison of Four Subjective Workload Rating Scales", Hill, Iavecchia, Byers, Bittner Jr., Zaklade and Christ identified that the Overall Workload (OW) and the National Aeronautics and Space Administration (NASA) Task Load Index (TLX) were consistently superior. Gawron (2000, p. 135) argues that the OW scale is "easy to use, but less valid and reliable" than the TLX scale. For this reason, the TLX scale was adopted in order to measure cognitive load. Paas et al. (2003) also suggests that performance can be measured, suggesting the variables of "correct test items, number of errors and time on task". The NASA TLX scale measures mental demand, physical demand, temporal demand, performance, effort and frustration. A copy of the task load index can be found in appendix 4.

3.4. Data Analysis

Both ratio variables (obtained through timing completion of task elements), and interval variables from each participant's NASA task load index questionnaire was analysed. Analysis was parametric; with group means compared using a two-sample t-test, to test the hypothesis. Data from each experiment type was compared. To test hypothesis A (comparison of touch with a supporting second hand) the two touch based experiments were evaluated. To test hypothesis B (comparison of mouse and touch) the mouse input was evaluated with the touch input method with the screen resting on a desk. For statistical significance the p threshold was determined as 0.05.

Timing data and responses from the task load index were compared separately in order to evaluate perceived difficulty and actual performance independently.

3.5. Limitations and potential problems

3.5.1. Reliability

The timing data was gathered electronically, and calculated by determining the time interval between start and end times (as opposed to using a timer). The ability for the test devices to update their time and date automatically was disabled to ensure the accuracy of the data. (This ensures that a device cannot synchronise with an external time server between the start and end of a measurement and so modify the end time recorded.) Data was transferred electronically from the test devices to the statistics software for the purposes of analysis. The full data set of electronically recorded data can be found in the CD located at appendix 6 in an anonymised form. The full source code of the testing software is available in in the appendices in digital form to enable readers to repeat the experiment if they so wish. Data from the task load index was entered manually at first and a reasonableness check (Lazar, Feng, & Hochheiser, 2010, p. 71) carried out. Later in the process the task load index was incorporated into the application so data could be recorded automatically. Scanned copies of the completed task load index survey can also be found in digitised form in the appendices, and data from the later participants is included within the CSV file of each test instance found within the digital media at appendix 6.

3.5.2. Validity

In order to preserve internal validity the experiment conditions were controlled as much as possible to minimise other variables. The test environments were kept free from distraction and background noise as much as possible, and the test was repeated to mitigate the effect of anomalous results (for example as a result of a participant sneezing during a task).

Whilst a non-random sample method of obtaining participants was utilised, it is recognised that until a population has grown up with both mouse and touch interfaces being in common use, it will be very challenging to identify a sample adult population without significantly more experience using a mouse controlled interface. So whilst the study can be generalised for the population of users familiar with touch screens, the research findings cannot be generalised for future populations.

As the participants were known to the researcher, the possibility exists that they may have, either consciously or subconsciously, wanted to bias the results of their experiment instance, however as the researcher had no vested interest in the results of the research leading to the rejection of the null hypothesis or otherwise, it was not possible for the participants to know what bias to adopt.

4. Results and Analysis

This chapter considers the results of the research experiment to ascertain whether, for each interaction method (selection, dragging and gesture) the data can lead to a rejection of either or both of the two null hypotheses in favour of the alternative hypothesis; with hypothesis A being that use of a touch-based interface with the secondary hand supporting the device affects the efficiency of interaction using a device situated on a desk, and hypothesis B stating that a touch-based interface for a system varies in its usability compared to an interface for the same system which requires use of a mouse. The chapter explains the data collection process, explores the findings from each task (selection, dragging and gesture) before considering the participants perceptions of the input methods obtained using the NASA task load index assessment.

4.1. The process of data collection

Data was collected over a 4-week period from 4th August to 2nd September 2014. Participants were members of staff and students from within the Department of Computer Science at the University of Chester and the Learning and Information Services department. All participants confirmed that they were experienced using a touch-based tablet computer device.
Initially only raw timing data was collected and exported from the application on the iPad and laptop in comma separated values (CSV) format, and data from the task load index collected in paper form. In order to facilitate easier data analysis the application was modified to find the mean average of timings for each of the three interaction events – selection, dragging and gesture. This in no way affected the method of data collection.

The task load index assessment, which participants completed upon completion of each task, was also incorporated into the application after the initial four participants had taken part. This was done in order to both speed the data collection process and removed the need for transcription (and potential for introduction of error). A screenshot of this can be found in appendix 5. It should be noted that some early participants indicated multiple consecutive points on the paper task load index (by drawing a ring on the scale encompassing multiple points), which was not possible on the electronic version of the form. When transcribing this data, the point closest to the centre of the marked selection was used.

Each set of raw data from each participant can be found within the digital media in appendix 6, alongside a Microsoft Excel document containing the aggregated means across all participants. Digital copies of scanned data can also be found within the digital media.
The evaluation software used to carry out the experiment was created using Xcode version 5.1.1, and should be able to run in this and later versions of Xcode for readers interested in evaluating the software or using it to carry out research of their own. This software can also be found on the digital media at appendix 6.

Four participants in the trial identified as being left handed, though only one of these participants used their left hand for both tasks, two participants used their left hand for the mouse and their right hand for interacting with the tablet, and one used their right hand for both tasks (in each case the choice to use the hand was out of personal preference).

The age ranges of the participants were recorded. Five participants were aged between 18 and 24, six aged between 25 and 34, two aged between 35 and 44, three aged between 45 and 54 and two aged between 55 and 65, there is therefore a slight bias towards the younger end of the adult population. There were eight female participants and ten male participants; gender was not linked to the data collected. No other demographic information was collected.

4.2. Data and Analysis

4.2.1. Selection Tasks

The selection data consisted of:

18 participants x

3 input methods x

5 selections x

8 repetitions = 2160 trials.

Mean and standard deviation values of the timing data for each of the selection input methods can be seen in Table 2 below, values are in seconds. The lowest mean selection times for any participant were as follows: Mouse: 3.185s, Desk: 1.825s and Held: 1.923s. Highest mean selection times were: Mouse: 5.926s, Desk 4.647s and Held 3.820s

	Mouse	Desk	Held
Mean	4.0713	2.7578	2.6327
Standard Deviation	0.7792	0.6631	0.5467

Table 2 - Mean selection times across input methods

Timing data for each participant collected was the total time for all five selections in each repeated instance and included the time required for participants to rectify errors (it was not possible to complete the task with uncorrected errors, and participants would need to continue to attempt the task until completion before progressing). Participants were given visual feedback as selection targets disappeared from the screen upon selection in the correct order. Participants were observed, and one of the eighteen participants was noted to use both hands during the selection task when the tablet was situated on the desk.

No unanticipated errors occurred during the selection tasks; errors were limited to participants initially pressing buttons in the wrong order, or missing the correct button. A column chart showing mean times for each task type can be seen in Figure 16 below.

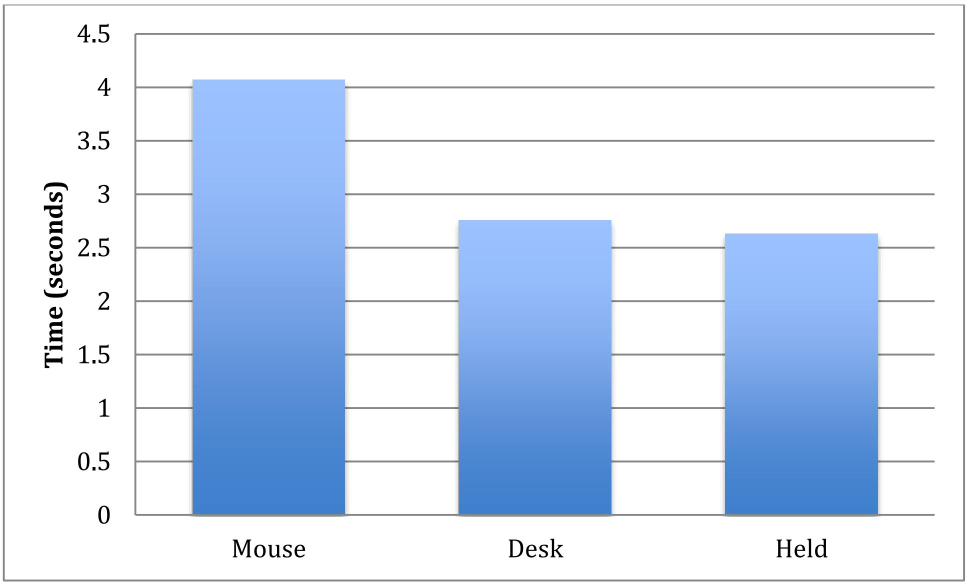

Figure 16 - Average time for selection tasks

4.2.1.1. Touch methods comparison – selection

Mean selection times indicate that selection is marginally slower when a touch screen is held by the user compared to when the touch screen is situated on a desk, with a difference in means of 0.12 seconds, however the results of a two tailed t-test indicates that there is low statistical significance in this result ($p=0.18$), so we are unable to reject the null hypothesis A for selection.

4.2.1.2. Mouse and Touch – selection

When comparing interaction with the mouse compared to touch on the desk situated iPad mean selection times indicate that selection is significantly quicker when using a touch screen, with a difference in mean of 1.31 seconds. At the 95% confidence level the confidence interval is from 0.94s to 1.69s (Mouse SE =0.18, Desk SE=0.15). This data indicates a 32% improvement in speed over interaction with the mouse.

Analysis using a two-tailed t-test of the mouse and desk based touch device data yields a p value of <0.000001 and $t=7.4270$, demonstrating statistical significance.

The finding that performance is increased when completing selection tasks using a touch based device compared to using a mouse is in line with the findings of Sears & Schneiderman (1991) and Cockburn, Ahlström, & Gutwin (2012).

For <u>selection</u> tasks, and as expected, the evidence therefore supports the rejection of the null hypothesis B in favour of the alternate hypothesis B (that touch varies the usability of a system compared to using a mouse), with the indication being that touch provides an improved user experience.

4.2.2. Dragging Tasks

The dragging task data consisted of:
18 participants x
3 input methods x
4 drag directions (each corner to diagonally opposing corner) x
8 repetitions = 1728 trials.

Mean and standard deviation values of the timing data for the drag task across each input method can be seen in Table 3 below, values are in seconds. The lowest mean drag time across all participants was as follows: Mouse 0.960s, Desk 0.807s, and Held 0.836s. The highest mean drag times were: Mouse 1.920s, Desk 1.332s, and Held 1.346s.

	Mouse	*Desk*	*Held*
Mean	*1.2470*	*1.000*	*1.0147*
Standard Deviation	*0.2758*	*0.1438*	*0.1562*

Table 3 - Mean drag times across input methods

Timing data collected included the time taken for each individual direction of the drag task from which the mean average was then calculated. Time data included the time required for participants to rectify errors (i.e. it was not possible to complete the task with uncorrected errors) Participants were given visual feedback in the form of dragged objects disappearing from the screen upon being correctly positioned within the target area.

One unanticipated error occurred during drag tasks for two separate participants, whereby the participants' fingers (or mouse pointer) moved passed the edge of either the top or bottom of the screen display area to the bezel (or simulated device bezel) and then returned to the display area, activating the iPad (or simulated iPad)'s notification or control centre, which then had to be swiped off the screen to reveal the task screen again. As each participant completed 32 drag tasks for each input method, the effect of this error will be minimal.

The aggregated average across all repetitions was then calculated, and can be seen for each input type in the column chart in Figure 17

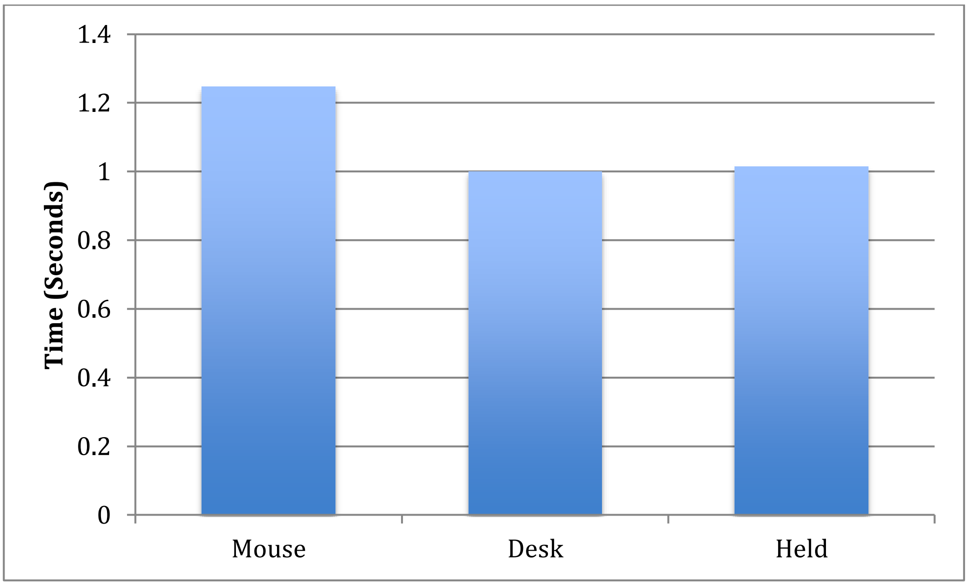

Figure 17 - Average time for dragging tasks

4.2.2.1. Touch methods comparison – dragging

Mean times for the dragging tasks indicate that drag based tasks are marginally quicker when the touch device is situated on a desk compared to being held; the difference in means being 0.015 seconds, less than 2% difference, and there is no evidence of statistical significance in this result (p=0.47, t=0.75). For dragging tasks, rejection of the null hypothesis A is not possible.

4.2.2.2. Mouse and Touch methods comparison – dragging

Mean times for the drag tasks between mouse and desk based tasks indicate that drag tasks are slightly quicker when completed using a touch interface compared to a mouse with a difference in mean of 0.25 seconds (a 20% improvement in speed), (Mouse SE=0.07, Desk SE=0.03). The difference is statistically significant with $p = 0.0002$ and $t = 4.7309$), and at the 95% confidence level the confidence interval is from 0.10s to 0.40s.

For dragging tasks the evidence supports rejection of the null hypothesis B in favour of the alternative hypothesis, with the indication that touch offers enhanced usability over that of a mouse.

4.2.3. Gesture Tasks

The gesture task data consisted of:
18 participants x
3 input methods x
1 drag task x
8 repetitions = 432 trials.

Mean and standard deviation values of the timing data for the gesture task across each input method can be seen in Table 4 below, values are in seconds. The lowest mean times for a participant to complete the gesture task were as follows: Mouse 1.853s, Desk 1.414s and Held 1.432s. The highest mean times were: Mouse 4.733s, Desk 1.332s and Held 3.348s.

	Mouse	*Desk*	*Held*
Mean	*3.1532*	*2.1752*	*2.1132*
Standard Deviation	*0.8415*	*0.5996*	*0.5216*

Table 4 – Mean gesture times across input methods

Timing data collected was the time taken for the user to scroll through the list and select the required item from which the mean average across all iterations and participants was then calculated. The time started with the first interaction with the list. Time data included the time required for participants to rectify errors (i.e. it was not possible to complete the task with uncorrected errors, participants had to locate and select the correct row), participants were given feedback whereby the programme progressed to the next screen once the correct item in the list was selected, or the incorrect row became highlighted when selected. No unanticipated errors occurred.

The aggregate average across all repetitions was then calculated, and can be seen for each input type in the column chart in Figure 18

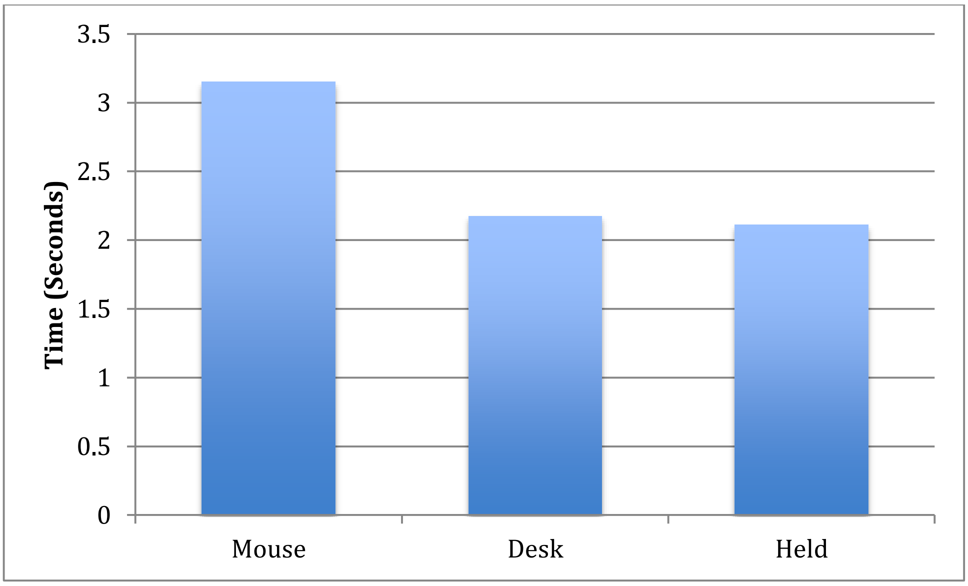

Figure 18 - Average time for gesture task

4.2.3.1. Touch methods comparison – Gestures

Mean times for the gesture tasks indicate that they are marginally slower when carried out on touch devices compared to being held, the difference in mean is 0.062 seconds (less than 3% difference), and there is no evidence of statistical significance in this result ($p=0.74$, $t=0.33$). For gesture-based tasks, the evidence does not support the rejection of the null hypothesis A.

4.2.3.2. Mouse and Touch methods comparison – Gestures

Mean times for the gesture-based tasks between mouse and desk based touch interaction indicate that gesture tasks are significantly quicker when completed using a touch interface compared to a mouse with a difference in means of 0.98 seconds (a 31% improvement in speed), (Mouse SE=0.14, Desk SE=0.20). The difference is statistically significant with $p = 0.0003$ and $t = 4.0162$).

For gesture tasks the evidence supports rejection of the null hypothesis B in favour of the alternative hypothesis, with touch again offering enhanced usability.

4.2.4. Perception

The values for perceived task load data are shown in Table 5. The values for perceived task load that could have been selected by the participants lie on a scale with a minimum value of 1 and maximum value of 21, and the mean average taken from across the 6 subcategories was used to calculate the group mean, and from that data the standard deviation was calculated.

	Mouse	Desk	Held
Mean	7.00	5.04	5.29
Standard Deviation	3.53	3.44	3.51

Table 5 - Perceived effort - mouse and desk based touch device

Figure 19 below shows the perception of each of the six categories covered by the NASA task load index survey.

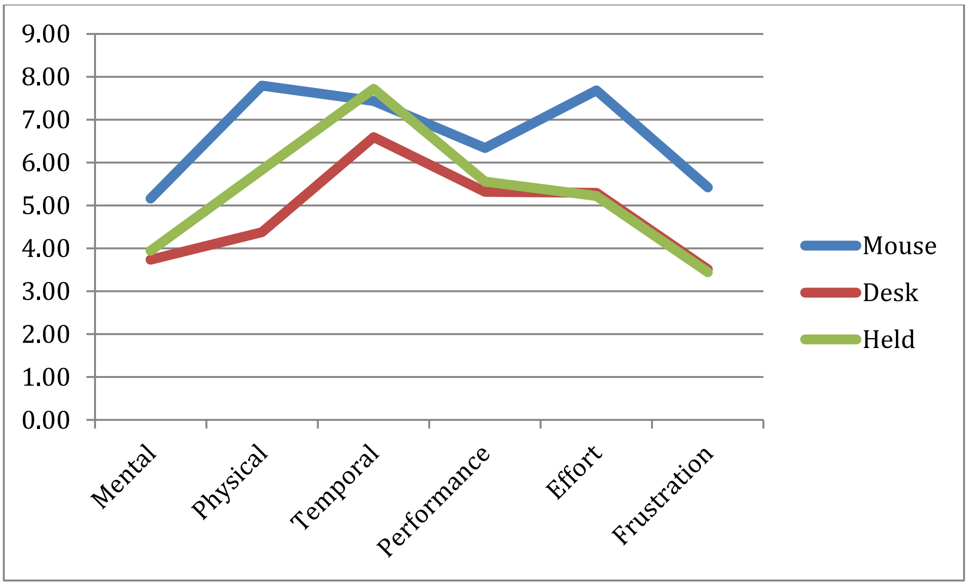

Figure 19 - Perceived task load

Participants regarded the mouse as being more mentally and physically demanding, as well as requiring more effort and inducing increased frustration – and for comparison of each of these measurements (mouse and desk values) the findings can be regarded as statistically significant with p values all <0.01.

For touch use, mean values indicate participants judged their performance to be similar, mental, physical and temporal demand when holding the iPad was deemed to be increased compared to using it on the desk, whereas effort and frustration were marginally lower, however opinions were significantly more varied among participants, and as such these values cannot be considered statistically significant.

4.2.4.1. Touch methods comparison – user perception

Analysis of perception data for both methods of touch input shows a difference in means of 0.25, with the held device requiring slightly more effort, though a two-tailed t-test indicates low significance of this difference in mean (p=0.68). Even if only the physical effort aspect of the task load index data is considered, which one might consider likely to show variation due to the effort required to hold the device, the difference in mean is only 1.2 and statistically insignificant (p=0.22).

4.2.4.2. Mouse and Touch methods comparison – user perception

Comparing the perception of mouse based interaction with that of the iPad located on the desk, the data indicates that that participants perceived that the selection tasks carried out on a touch screen required less effort than using a mouse (higher values indicate greater perceived requirement of effort). A two-tailed t-test yields values of $p = 0.0006$, $t = 4.23$, indicating strong statistical significance. This again supports the rejection of null hypothesis B in favour of the alternate when considering user perception across multiple forms of input, with the indication being that a touch interface is perceived to have a lower task load than a mouse controlled interface.

5. Discussion and Conclusion

In summary the results indicate that for the three forms of interaction, touch input outperformed that of mouse input, but there is no evidence to suggest that holding the iPad has any effect on the efficiency of touch.

5.1.1. Selection

In Sears and Shneiderman's (1991) paper, with the largest target size (32 pixels square) and comparing the data from the non stabilised touch screen, they observed a 40% increase in selection time when using direct touch compared to the mouse, slightly higher than the 32% observed in this experiment. It should be noted that they indicate there was no mouse acceleration used: "mouse was calibrated … so a single pass horizontally on the pad resulted in the cursor moving the width of the screen" (p. 5). Cockburn et al.'s (2012, p. 225) study indicates a 34% increase in speed, a very similar value to this experiments results, despite having a cursor offset from the touch position. However, they excluded errors from their analysis, which they note were higher for touch input (they do not indicate whether mouse acceleration was enabled). For selection tasks the findings of this experiment reinforce the findings of both Sears & Shneiderman's and Cockburn et al.'s research.

5.1.2. Dragging

Calculations on Forlines et al. (2007) experiment data identified that touch input was on average 18% slower than mouse input for dragging tasks (their paper refers to docking, but in both instances the time taken to move an object into a target area is considered). In Forline et al.'s experiment the object was required to be within 5 pixels of the target which Cockburn et al. (2012, p. 221) note to be "unusual in desktop interaction". Cockburn et al.'s later study (2012, p. 225), which used a cursor offset from the finger and varying sizes of object width (5mm to 200mm) and distance (50mm to 200mm), found that touch input was only 10% slower than the mouse input for dragging tasks in the left to right direction, but participants in their study were not frequent users of a touch screen. Data from this experiment (which had a drag distance of 85mm, dragging tasks in multiple directions, and a drag object width of 39mm, height of 23mm) shows a 20% increase in speed for touch input compared to mouse.

Two hypotheses may be generated from these findings: firstly that given a large enough object for selection with proportional target area, usability for touch interface will exceed that of a mouse. There is evidence for this in Cockburn et al.'s study (2012, p. 226), where they plot the index of difficulty (based on object size and distance dragged) and the values for mouse and touch converge at a lower index of difficulty. At the lowest index of difficulty they measure (2.32 bits), task completion time is almost the same. Extrapolating Cockburn et al.'s graph shown in Figure 20 below, the slightly lower index of difficulty used in this experiment of 2.26 (using the Shannon formulation of Fitts's law), we might expect to see a marginally lower time for touch, but not of the magnitude of 20% that has been identified.

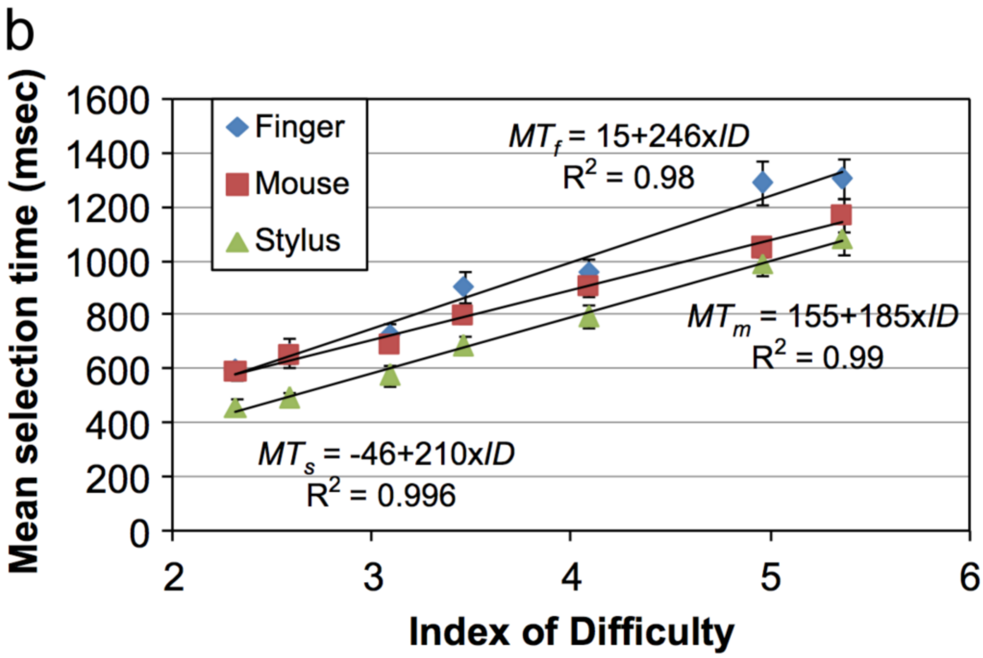

Figure 20 - Cockburn et al. (2011)'s dragging task results

The user interfaces evaluated in the review of literature made use of larger targets for selection and we can infer that for elements that may be dragged, the use of larger user interface components should also be beneficial.

The second hypothesis that may explain the larger than anticipated improvement in usability for touch over mouse in dragging tasks is that having previous experience of touch screen devices (as participants in this experiment had, but participants in earlier studies did not) enhances usability, though further study would be needed to prove causation.

5.1.3. Gestures

There is limited research into the effectiveness of gestures, particularly when comparing mouse and touch interfaces, although some work has been done exploring the use of mice with additional sensors to detect gestures using multiple touches, rather than relying on gesture detection using a standard mouse (Villar, et al., 2009), such technology has since been implemented and can be found in Apple's mighty mouse (Apple Inc., 2014). Whilst this experiment indicates that gestures performed on a touch screen are more effective than a mouse, the mouse hardware did not permit the use of gestures performed by one or more fingers on the mouse surface, rather it relied on the user to apply the gesture using a 'click-drag-release while dragging' motion. It is hypothesised that results would differ had the mouse used been able to detect a swipe gesture across the mouse's surface.
The Windows 8 operating system makes use of gestures, and commentators have noted that it is easier to use with a touch screen than a mouse (Branscombe & Grabham, 2013), or with a mouse that can detect gestures upon its surface (Shultz, 2012) which would support the findings of this experiment.

5.1.4. Effectiveness of one hand supporting the device

Comparison of all the user interface interaction methods (touch, drag and gesture) yielded no evidence that supporting the device provided any advantage over that of having the device located on a desk, despite evidence identified during the review of literature that the second hand can provide benefit to direct touch interaction and in support of a task being carried out by the primary hand, which was unexpected.

Participants were observed carrying out the tasks and it was noted that many participants used the desk in front of them, their leg, or the arm of the chair they were sat in to support the arm and wrist of the hand holding the iPad; this had the effect of restricting the movement of the supporting hand. Some participants commented that the device was heavy, and one mentioned that he could hold his own personal device at both edges with one hand, which he was unable to do with the testing device. It is possible that these factors contributed to the result, however it would require further study to determine their impact. Of those participants that held the iPad with the wrist and arm unsupported, varying degrees of movement of the supporting hand were observed.

5.2. Limitations

Perception data was only collected upon completion of all tasks and was not specific to each of the three interaction methods (selection, dragging and gesture), this limits the validity of the data to input methods alone, and it is possible that because participants were unlikely to have performed gestures using a mouse before, it may be biased in favour of touch devices.

When participants undertook the gesture task they were required to click and drag to scroll through the list (although the software implemented the acceleration of this list based on mouse acceleration). More commonly lists in a user interface operated using a mouse can usually be controlled using a mouse's scroll wheel where present, or indeed items in the list can be highlighted through use of the device's keyboard, which was not an option in this study as it was solely concerned with comparison of touch and mouse.

In Sears and Shneiderman's (1991) experiment the mouse was configured so that moving the cursor the width of the screen corresponded to moving the mouse the full width of the mouse mat. Modern computer operating systems allow the mouse sensitivity to be adjusted, and further provide additional features such as acceleration, so quicker movements cover more of the screen and slower movements result in less movement and hence greater accuracy. Whilst participants were not permitted to use a mouse mat, and had a large area of desk upon which they could operate the mouse, in order to more fairly test mouse use, the mouse element of the study should be optimised for each individual user, for example though the use of a preferred mouse and opportunity to set tracking and acceleration speeds that best suit the user.

Fatigue was another potential issue in this research, as the experiment required the user to hold the tablet. Even though this experiment required the user to hold the tablet for a relatively short period of time some users reported fatigue, over a longer period of time the weight of a tablet computing device is likely to have a greater fatiguing effect on the user, though newer lighter devices will mitigate this effect to some extent.

5.3. Conclusion

5.3.1. Research Objectives: Summary of Findings and Resulting Conclusions

The aim and objectives were as follows:

- *To compare the effectiveness of completing identical tasks when using a touch screen compared to mouse input*. The findings indicate that with relatively large targets, touch is more effective then mouse input for selection, dragging and gesture tasks.
- *To determine the perceived difficulty of tasks performed on both a touch interface and mouse interface*. The findings indicate that with relatively large targets, users perceive touch based selection, dragging and gesture tasks to require less effort than when using a mouse.
- *To determine if the bimanual use of a touchscreen (with one hand in a supporting role) elevates its usability above that of interaction using a single hand.* The experiment found no evidence that this is the case.

5.4. Contribution to Knowledge

The review of literature indicated a possible flaw in the selection of participants for previous studies when using a touch screen, in that they had no prior experience with such devices, although in many cases the lack of pervasion of touch screen technology was the limiting factor, not the research design. This research has demonstrated that, unlike previous findings, tasks involving dragging, with a large enough user interface elements size, are more effectively carried out by touch compared to using a mouse when participants have prior experience with a touch screen. These findings help in understanding the true effectiveness of touch interfaces, and the benefits of using suitably large user interface components when developing user interfaces for touch control.

5.5. Self-reflection

This research work started with a different intention. To develop software for a touch screen device to facilitate the display of lyrics on a separate screen (for example for houses of worship or karaoke) replicating functionality already available on software designed for operation using a keyboard and mouse. Only after a significant amount of work, including the development of prototype software and a partial literature review did the researcher realise that not only would an accurate comparative evaluation be too challenging to carry out using the resources available to him, but also that the overall research goal lacked focus, despite the researchers significant interest in software development for touch devices. Most of the work undertaken up to this point was therefore abandoned. It was however through carrying out the literature review that the researcher identified the subject for the research eventually carried out, which is hoped will have benefits for both a wider community of academics, user interface developers and eventually end users of computing devices. The researcher also had to undertake significant study into identifying and using appropriate forms of statistical analysis in order to properly evaluate the data generated by the experiment, and this knowledge gained will be of significant benefit for any future research undertaken.

5.5.1. Recommendations and suggestions for further study

5.5.1.1. Target size

In order to fully understand the benefits of large user interface elements, and in particular the point at which usability for touch exceeds that of a mouse, further study should be undertaken with different target sizes, and for drag tasks with different drag distances. Using Fitts's law a model could be developed to identify the optimal element sizes for user interfaces which can be controlled by both mouse and touch; this would have uses in, for example, developing the user interface of a website which shares the same interface on tablet and desktop computing.

5.5.1.2. Familiarity with touch devices

Further study should also be undertaken to identify the effect of prior familiarisation with touch device, and potentially mouse devices. This could for example involve undertaking a longitudinal study with participants who are current mouse users but have minimal or no touch screen experience, by running the experiment, allowing them to become familiar with touch screens over time, and repeating the experiment at set intervals. Furthermore, studies could be undertaken with users who are using both mouse and touchscreens for the first times, for example school children, exploring whether there is a variation in user interface interaction and perception compared to that found in adults who were exposed to mouse based interface before touch interfaces. Particular consideration should be given to ethical concerns for any research undertaken using child participants.

5.5.1.3. Bimanual interaction of touch devices

With regard to the use of one hand to support the device, it would be worthwhile pursuing further study to determine whether the weight and size of the device are a preventative factor in the second hand enhancing user input effectiveness, or if the utility of a supporting second hand is something that may be beneficial when 'learned' by the user. Movements made by the second hand could be studied in detail using an iPad's accelerometer and gyroscope.

5.5.1.4. Drag direction and handedness

The direction of drag gestures and the handedness (right or left handed status) of users could also be investigated. Cockburn's study (2012, p. 227) which additionally considers radial dragging tasks of solely right handed users found that drag tasks in the north westerly direction showed a higher occurrence of errors and were more slowly completed, particularly when using touch.

Reference List

Apple Inc. (2013a). *iPad - View all the technical specifications*. Retrieved July 18, 2013 from Apple (United Kingdom): http://www.apple.com/uk/ipad/specs/

Apple Inc. (2013b). *Macbook Air - Technical Specifications*. Retrieved July 18, 2013 from Apple (United Kingdom): http://www.apple.com/uk/macbook-air/specs.html

Apple Inc. (2013c, January 28). *Event Handling Guide for iOS.* Retrieved July 18, 2013 from iOS Developer Library: http://developer.apple.com/library/ios/#documentation/EventHandling/Conceptual/EventHandlingiPhoneOS/multitouch_background/multitouch_background.html

Apple Inc. (2013d). *GarageBand*. Retrieved July 11, 2013 from Play and record music with GarageBand: http://www.apple.com/uk/apps/garageband/

Apple Inc. (2014). *Magic Mouse*. Retrieved September 4, 2014 from Apple: https://www.apple.com/magicmouse/

Atmel. (2013). *Atmel maXStylus*. Retrieved July 10, 2013 from Atmel maXStylus: http://www.atmel.com/microsite/stylus/default.aspx

Beck, F., & Stumpe, B. (1973). *Two devices for operator interaction in the central control of the new CERN accellerator.* Geneva: CERN.

Biggam, J. (2011). *Suceeding with your master's dissertation: a step-by-step handbook* (2nd Edition ed.). Maidenhead, England: Open University Press.

Blanch, R., Guiard, Y., & Beaudouin-Lafon, M. (2004). Semantic pointing: improving target acquisition with control-display ratio adaptation. *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 519-526). Vienna: ACM.

Bragdon, A., Nelson, E., Li, Y., & Ken, H. (2011). Experimental Analysis of Touch-Screen Gesture Designs in Mobile Environments. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 403-412). Vancouver: ACM.

Branscombe, M., & Grabham, D. (2013, March 25). *Windows 8 review.* Retrieved September 12, 2014 from TechRadar: http://www.techradar.com/reviews/pc-mac/software/operating-systems/windows-8-1093002/review/7

Bravo, R. Z. (2013). *Decriptive Study on the Use of Bimanual and Same-hand Multifinger Interaction on a Multitouch Display.* Retrieved May 27, 2014 from http://www.diva-portal.org/smash/get/diva2:644458/FULLTEXT02

British Computer Society. (2004, September 1). *The British Computer Society Code of Good Practice.* Retrieved June 26, 2014 from British Computer Society: http://www.bcs.org/upload/pdf/cop.pdf

CERN. (2010, March 16). *Another of CERN's many inventions!* Retrieved July 18, 2013 from CERN Document Server: http://cds.cern.ch/record/1248908

Chin, R. M. (2010). *Patent No. 2012/0050176.* US.

Cockburn, A., Ahlström, D., & Gutwin, C. (2012). Descriptive Study on the Use of Bimanual and Same-hand Multifinger Interaction on a Multitouch Display. *International Journal of Human-Computer Studies , 70* (3), 218-233.

Fitts, P. M. (1954). The Information Capacity of the Human Motor System in Controlling the Amplitue of Movement. *Journal of experimental psychology , 47* (6), 381.

Forlines, C., Wigdor, D., Shen, C., & Balakrishnan, R. (2007). Direct-Touch vs. Mouse Input for Tabletop Displays. *Conference on Human Factors in Computing Systems* (pp. 647-656). San jose, CA: ACM.

Gawron, V. J. (2000). *Human Performance Measures Handbook.* Mahwah, New Jersey: Laurence Erlbaum Associates.

Guiard, Y. (1987). Asymetric Division of Labor in a Human Skilled Bimanual Action: The Kinematic Chain as a Model. *Journal of Motor Behavior , 19* (4), 486-517.

Hill, S. G., Iavecchia, H. P., Byers, J. C., Bittner Jr., A. C., Zaklade, A. L., & Christ, R. E. (1992). Comparison of Four Subjective Workload Rating Scales. *Human Factors: The Journal of the Human Factors and Ergonomics Society , 34* (4), 429-439.

Johnson, E. (1967). Touch Displays: a Programmed Man-Machine Interface. *Ergonomics , 10* (2), 271-277.

Lazar, J., Feng, J. H., & Hochheiser, H. (2010). *Research Methods in Human-Computer Interaction.* Chichester, England: Wiley.

Love, S. (2005). *Understanding Mobile Human-Computer Interaction.* Burlington, MA: Elsevier.

Microsoft. (2014). *Change Mouse Settings - Windows Help.* Retrieved July 14, 2014 from Windows: http://windows.microsoft.com/en-us/windows/change-mouse-settings

Microsoft Corporation. (2013). Microsoft® PowerPoint® 2013 [Computer Software]. Redmond, Washington: Microsoft.

Microsoft Corporation. (2014). *Microsoft Remote Desktop (version 8.0.7) [Mobile application software].* From https://itunes.apple.com

Mignonneau, L., & Sommerer, C. (2005). Designing emotional metaphoric, natural and intuitive interfaces for interactive art, udutainment and mobile communications. *Computers and Graphics , 29* (6), 837-851.

Mitchell, M. L., & Jolley, J. M. (2013). *Research Design Explained* (8th Edition ed.). Belmot, CA: Wadsworth, Cengage Learning.

National Museum of American History. (2013). *Overhead Projectors.* Retrieved August 7, 2013 from National Museum of American History: http://americanhistory.si.edu/mobilizing-minds/overhead-projectors

Neilsen, J. (1993). *Usability Engineering.* Cambridge, MA: AP Profesional.

Nielsen, J. (1995, January 1). *10 Usability Heuristics for User Interface Design.* Retrieved August 7, 2013 from Nielson Norman Group: http://www.nngroup.com/articles/ten-usability-heuristics/

Nielsen, J. (2012, January 4). *Usability 101: Introduction to Usability.* Retrieved September 12, 2014 from Nielsen Norman Group: http://www.nngroup.com/articles/usability-101-introduction-to-usability/

Paas, F., Tuovinen, H. T., & Van Gervan, P. W. (2003). Cognitive load measurement as a means to advance cognitive load theory. *Educational Psychologist , 38* (1), 63-71.

Pogue, D. (2013). Out with the Real. *Scientific American , 308* (2), 29.

Sears, A., & Shneiderman, B. (1991). High precision touchscreens: design strategies and comparisons with a mouse. *International Journal of Man-Machine Studies , 34* (4), 593-613.

Shultz, G. (2012, November 15). *Discover the benefits of the Microsoft Touch Mouse in Windows 8.* Retrieved September 12, 2014 from TechRepublic: http://www.techrepublic.com/blog/windows-and-office/discover-the-benefits-of-the-microsoft-touch-mouse-in-windows-8/

Subramanya, S. R., & Yi, B. K. (2006). User interfaces for mobile content. *Computer , 39* (4), 85-87.

Villar, N., Izadi, S., Rosenfeld, D., Benko, H., Helmes, J., Westhues, J., et al. (2009). Mouse 2.0: Multi-touch Meets the Mouse. *Proceedings of the 22nd annual ACM symposium on User interface software and technology* (pp. 33-42). Victoria: ACM.